面向移动终端的云监控研究

徐海浪 袁家斌
(南京航空航天大学计算机科学与技术学院 南京 210016)

摘 要 针对移动终端对病毒防治的高效率和轻量级需求,运用云安全技术对主机入侵防御系统(HIPS)进行改进,形成一种云监控模型。增加文件判断功能、将规则库和文件判断工作移至云端服务器,降低了系统占用,轻量化了客户端;改变规则制定策略,针对不同病毒制定不同规则,降低了规则的复杂性,提高了规则匹配效率;通过黑白名单技术和单步危险行为分析法,降低了客户端与服务器的通信代价,提高了文件判断效率;改变系统监测模式,变主动监控为被动监控,降低系统监测的工作时间,提高了云监控模型的工作效率。最后通过形式化方法证明了云监控模型的安全性。

关键词 主机入侵防御系统,移动终端安全,实时监控,云安全
中图法分类号 TP309.5 **文献标识码** A

Research on Cloud Monitoring Oriented to Mobile Terminal

XU Hai-lang YUAN Jia-bin
(College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, Nanjing 210016, China)

Abstract Aiming at the demand of the high efficiency and lightweight client of virus prevention for mobile terminal, this paper improved the HIPS by the technology of cloud security to form the cloud monitoring model. Through adding file property judge function and moving rule library and the work of file property judge to the server, the server system occupies was reduced. Through changing the strategy of rule-making, according to different virus set different rules, the complexity of the rules was reduced and the efficiency of rules matching was improved. Through the black and white list technology and a single step dangerous behavior analysis, the cost of communication between the client and server was reduced and the efficiency of file property judge was improved. Through changing monitoring mode, changing the active monitoring to passive, the working time was reduced and the working efficiency of the cloud monitoring model was improved. Finally, the formal method proves the security of the cloud monitoring model.

Keywords HIPS, Mobile terminal security, Real-time monitoring, Cloud security

1 引言

移动终端以其便利、经济、持续在线性等优势占据了巨大的客户群体,是移动互联网发展的重要基石。随着3G时代的到来,移动终端功能增强,与用户关系更加密切,安全问题的重要性和迫切性逐步上升[1,2]。中国互联网协会反网络病毒联盟(ANVA)监测数据[3]显示,2010年移动终端病毒累计感染智能移动终端800万部,严重危害移动终端用户的信息安全。当前,移动终端的病毒防护采取的是传统的计算机病毒防治技术,但是相比于计算机,移动终端有限的处理能力、存储空间和电池容量,使得传统的病毒防治技术不能直接应用于移动终端,因此需要具有更轻量级、高效率的客户端的安全产品。

移动终端的病毒防治思路不外于借鉴计算机病毒的防治方法。目前的研究或者产品也正是以计算机病毒的研究为基础,或改进算法,或改变运行模式,来提高病毒检测效率,降低病毒检测消耗。文献[4]从优化扫描引擎出发,提出一种基于有序二叉树的特征码多模式匹配算法,其提高了病毒扫描速度,降低了对移动终端空间和性能的消耗;文献[5]将病毒检测的特征码技术和启发式技术相结合,在完全摈弃特征码匹配算法思路上进行了大胆探索;文献[6]分析了特征码扫描、启发式方法和行为监测3种病毒检测方法的优缺点,提出将启发式方法和行为监测技术相结合的病毒防治方案。文献[4-6]的研究虽然提高了病毒检测效率,在一定程度上降低了对移动终端的资源占用,但是它们仍是基于传统计算机的病毒防治模式,要么需要频繁更新、要么实现难度较大,这并不能满足移动终端的需求。文献[7]跳出传统思路,将病毒防治的分析系统运行于移动通信网络上,避免了频繁的更新,极大地降低了对移动终端的性能影响,但是对于通过内存卡、蓝牙、红外或者电脑数据线连接获得的软件无法进行安全分析。

云计算是网格计算、分布式计算、并行计算、虚拟化技术等和网络技术混合演进的结果,它将网络上的各种资源(软

到稿日期:2011-09-06 返修日期:2011-11-02 本文受基金项目(2009AA044601),国家863重大项目资助。
徐海浪(1985—),男,硕士生,主要研究方向为信息安全、云安全;袁家斌(1968—),男,博士后,教授,博士生导师,主要研究方向为信息安全、高性能计算、量子密码。

件、平台、基础设施等)作为一种服务向用户提供(用户可按需获取服务),能够为用户提供海量的存储空间和超级计算能力[8,9]。起源于云计算的云安全技术,采用典型的 CS(Client/Server)结构,将病毒防治的大部分计算任务移至服务器,作为一种云服务提供给用户,在提高病毒检测效率的同时有效降低了传统反病毒软件对系统性能的消耗[10,11]。以访问控制为基础的主机入侵防御系统(HIPS)是一种根据可定制的规则对系统中程序的运行、注册表的读写等进行允许或阻止的防御系统,可以同时防御已知病毒和未知病毒[12]。本文用云安全技术对 HIPS 进行改进,形成本文的云监控基本模型:增加规则智能制定并将制定工作移至服务器,作为一种云服务向用户提供,以解决 HIPS 易用性差的问题;增加规则列表并将规则库移至服务器,将规则作为一种云资源向用户提供,以提高规则匹配效率,满足移动终端对客户端的高效率和轻量级需求;改变系统监测模式,由系统中文件(包括应用程序、蓝牙等)启动系统监控功能,提高系统监测工作效率,进一步降低对移动终端的性能影响。

2 云监控基本模型

传统反病毒软件的系统监控和查杀是基于病毒特征码进行的,对于尚未分析出特征码的病毒是无法识别的。相对地,HIPS 是基于软件行为的,判断其是否违反了规则库中的规则,然后再据此做出反应。因此,HIPS 既可以防范已知病毒,也可以防范未知病毒,甚至正常程序的危险行为(加载驱动等),而 HIPS 的实现相对于复杂的沙盘等新一代病毒防治方法更为简单,对系统资源的占用也相对较少,比较适合于移动终端。

但是另一方面,HIPS 需要用户手动添加规则,对用户的专业水平要求高,如果用户的基础知识不足,错误放行,那么恶意程序就会侵入系统,因此其易用性较差,不适合移动终端的广泛用户群体。为解决 HIPS 的应用矛盾,增加规则制定功能,其由服务器智能生成或者后台专业人员制定,如图 1 所示。系统监测模块根据规则库中的规则监测系统中各种行为操作,规则库则通过与服务器的交互完成规则的更新。

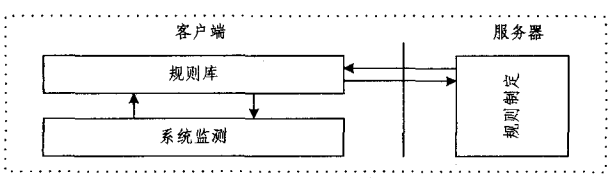

图 1 带规则自动生成的 HIPS 模型

传统的 HIPS 试图通过通用的规则来防御所有的病毒入侵,但是病毒种类繁多、行为千差万别,势必要求这些通用的规则足够复杂、多样,而且随着新病毒的不断出现,这些规则会像病毒特征库一样要求不断更新。我们改变传统的规则制定方法,增加文件性质判断功能来识别每一个文件的性质,并针对每一个病毒文件制定一条规则,同时利用云安全技术,将文件性质判断工作移至服务器进行,同时将规则库移至服务器,在客户端上只保存本地文件的访问规则,形成本文提出的云监控基本模型,如图 2 所示。规则列表规定了本地文件的访问规则,规则库中是服务器收集的所有文件的访问规则,文件判断的功能在于识别文件的性质(属于正常文件或病毒文件)。

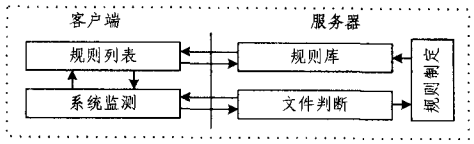

图 2 云监控基本模型

3 云监控工作流程

为了便于说明云监控模型的工作流程,进行如下形式化定义。

定义 1 文件性质集 $R\{r_1,r_2,r_3\}$ 表示文件的性质,分别为正常文件 r_1、病毒文件 r_2、可疑文件 r_3。

定义 2 文件集 $F\{f_1,f_2,f_3,\cdots,f_n\}$ 是所有已设定访问规则的文件集合,要么为正常文件 r_1,要么为病毒文件 r_2,包括系统文件、用户文件和应用程序等。

定义 3 文件集 $F'\{f_1',f_2',f_3',\cdots,f_{n'}'\}$,其中 $f'\in F$,是某一移动终端上已设定访问规则的文件集合,要么为正常文件 r_1,要么为病毒文件 r_2,包括系统文件、用户文件和应用程序等。

定义 4 规则集 $N\{n_1,n_2,n_3,\cdots,n_n\}$ 是文件访问规则的集合,$n_1=${进行文件性质判断}为默认规则,表示某文件属于未知性质的文件,需要进行文件性质判断。

定义 5 规则库 $M=\{(f,n)\mid f\in F,n\in N\}$ 是文件 f 的和规则 n 的二元组,表示文件 f 要遵守的访问规则。

定义 6 规则列表 $M'=\{(f',n')\mid f'\in F',n'\in N\}$,其中 $M'\subset M$,是移动终端上存在的文件的访问规则集合。

定义 7 文件判断函数 $k(f_i)=r_j$ 是从文件 f_i 到文件性质 r_j 的一个映射。

定义 8 规则生成函数 $h((f_i,r_j))=(f_i,n_k)$,规则制定模块根据文件角色 r_j 生成默认的规则 (f_i,n_k)。

定义 9 规则库更新函数 $y((f_i,n_k))=M\cup\{(f_i,n_k)\}$。

下面对云监控模型的具体工作流程进行说明,工作流程如图 3 所示。

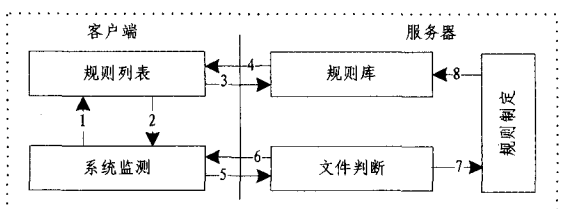

图 3 云监控模型工作流程

系统监测模块监控系统运行,当发现文件 f 运行时:

1. 系统监测模块查询规则列表 M',查询与文件 f 相对应的访问规则,当规则列表中存在文件 f 的访问规则即 $f\in F'$ 时,转向第 2 步,当规则列表中不存在文件 f 的访问规则即 $f\notin F'$ 时,转向第 3 步;

2. 规则列表向系统监测模块返回文件 f 的访问规则 (f,n_i),系统监测模块接收到访问规则后,按照访问规则监控文件的运行,直到文件运行结束;

3. 客户端向服务器的规则库 M 查询文件 f 的访问规则;

4. 服务器从规则库中查询文件 f 的访问规则,当规则库中存在文件 f 的访问规则即 $f\in F$ 时,返回规则 (f,n_j),其中 $j\neq 1$,转向第 2 步,当规则库中不存在文件 f 的访问规则即

$f \notin F$时，返回默认访问规则(f,n_1)，其中n_1={进行文件性质判断}，转向第5步；

5. 系统监测模块按照规则(f,n_1)，向文件判断模块提交文件性质判断请求，并提供相关文件信息，文件判断模块进行文件性质判断$r=k(f)$，当文件判断结果为可疑文件r_3时，转向第6步，当文件判断结果为正常文件r_1或者病毒文件r_2时，转向第7步；

6. 要求系统监测模块提供进一步的文件信息，转向第5步；

7. 将文件信息和文件性质交与规则制定模块，由规则制定模块制定访问规则$(f,n')=h((f,r))$；

8. 规则制定模块更新规则库$M=y((f,n'))$，即$M=M\cup\{(f,n')\}$，同时将规则n'加入到规则集N中，得到$N=N\cup\{n'\}$。

云监控模型的功能在于对系统中的文件按照相应的访问规则进行监控。根据上述工作流程，可以得出云监控的功能函数：

$$Q(f)=\begin{cases}(f,n_i), & \text{当 } f\in F' \text{时}\\ (f,n_j),\text{且 } j\neq 1, & \text{当 } f\notin F' \text{但 } f\in F \text{时}\\ (f,n')=h((f,k(f))), & \text{当 } f\notin F \text{时}\end{cases}$$

当规则列表中已有f的访问规则即$f\in F'$时，$Q(f)=(f,n_i)$；当规则列表中没有对应的访问规则即$f\notin F'$，而$f\in F$即f为已知的正常文件或者病毒文件时，$Q(f)=(f,n_j)$，且$j\neq 1$；当$f\notin F$即f为可疑文件时，$Q(f)=(f,n')=h((f,k(f)))$。

4 云监控实现机制

4.1 规则列表与规则库

从云监控的工作流程可以看出规则列表和规则库需要具有对文件f进行性质判断的功能，给出文件f属于正常文件r_1、病毒文件r_2或者可疑文件r_3的结论，而传统的特征码检测方法只能给出是否是已知病毒的二值判断，不能满足现在的需求。黑白名单技术通过对黑白名单的管理，实现对目标的三重分类：属于黑名单、属于白名单、既不属于黑名单也不属于白名单，通过合理的黑白名单设置，可以完成规则列表和规则库的功能需求。黑名单包含已知病毒文件的标识，白名单包含已知正常文件的标识，客户端向规则库查询文件性质时，需要提供文件的标识，当与黑名单中文件标识匹配时，说明是一个病毒文件，当与白名单中文件标识匹配时，说明是一个正常文件，当都不匹配时，说明是一个可疑文件。

黑名单技术的关键是文件标识的选取，需要避免使用文件名、文件长度等易被绕过的标识。规则库处于服务器中，对文件进行性质判断需要文件信息，若将文件直接上传，会造成系统负担，而且对于大文件上传会造成实时性降低、网络堵塞等问题，这也是规则库采用黑白名单技术，而不使用特征码进行检测的原因。消息摘要算法(MD5)是计算机安全领域广泛使用的一种散列函数，用以提供消息的完整性保护，具有很好的效果。因此本文采用MD5值作为文件的唯一标识，在文件性质判断中代替文件自身上传至服务器。

规则库具有完整的正常文件和病毒文件信息，规则列表只包含本地文件的信息，降低了对移动终端的存储空间占用，减少了客户端进行已知病毒判断时对移动终端的性能消耗。规则列表的更新只有在移动终端中出现新的文件时才会进行，这就降低了客户端与服务器之间的通信频率，减少了更新消耗。

4.2 系统监测

系统监测作为云监控模型的功能核心，是云监控几大功能模块中最消耗移动终端资源的模块。传统的系统监测思路是保持系统监测时刻处于运行状态，时刻监测系统中文件的变化(运行、修改等)，然后按照对应的访问规则进行监控，这可以理解为主动监控。主动监控因为系统监测模块时刻处于运行状态，对系统的性能影响及资源消耗较大，工作效率较低。为降低系统监测对系统的性能影响，将主动监控改为被动监控，即当系统接受到文件(应用程序、蓝牙、网络模块等)启动请求时，首先检测其在规则列表中的访问规则，当属于需要监控的文件时，启动系统监测模块，然后再启动发起请求的文件。这样，云监控最消耗资源的模块只有在需要的时候才会被启动，节约了资源，提高了工作效率。

如果用$F(X)$表示某一功能文件(应用程序、蓝牙等)，不妨假设$F(X)$的启动过程为：

```
{
  主体发起启动 F(X)请求;//主体可以是系统或者用户
  系统检测 F(X)的启动条件;
  系统启动 F(X);
}
```

只要将此过程增加两条语句就可以实现被动监控功能：

```
{
  主体发起运行 F(X)请求;
  系统检测 F(X)在规则列表中的访问规则;//增加的语句 1
  if(F(X)需要进行监控) //增加的语句 2
{
    启动云监控系统监测模块;
}
  系统检测 F(X)的启动条件;
  系统启动 F(X);
}
```

为保证监控的安全性，在规则列表中需要将能够与外界进行信息交互的功能模块(比如蓝牙、网络连接、移动存储介质接口等，它们本身属于正常文件范畴)都加入到规则列表中，并作为需要监控运行的文件制定相应的访问规则，如此当系统与外界发生信息交互时，云监控系统监测模块必然处于运行状态。

4.3 文件判断

文件判断模块需要实现对规则库不能判断的文件进行性质判断，给出正常文件r_1、病毒文件r_2或者可疑文件r_3的结论。对于目标文件的未知判断(即文件不属于已知的正常文件或者病毒文件)，目前有启发式行为分析法、沙盘、虚拟执行等方法，考虑到移动终端的性能特点和客户端、服务器之间的通信代价，采用行为分析方法。由系统监测模块监测并上传文件在移动终端上的行为操作，由服务器上的文件判断模块判断文件的性质。

采用行为分析方法判断文件的性质，一方面，客户端需要不断上传文件行为，使得客户端和服务器通信频繁，增加了系统通信代价；另一方面，行为分析依据文件行为判断文件的性质，做出正常文件的判断需要全部文件行为，但是病毒性判断

并不总是需要完整的文件行为，可能只需要某一部分行为就可以做出判断。病毒文件必然会对系统造成影响，因此或者修改系统文件，或者根据其他应用程序文件进行更改，或者根据用户文件操作的特点引入危险行为的概念，提出单步危险行为分析方法，即：系统监测模块监测文件的运行，并记录文件每一个动作，当出现危险行为时，将记录的行为上传至服务器，由文件判断模块进行分析。危险行为是指会对系统文件、其他应用程序文件、用户文件造成更改的行为，如表1所列。这种方法能够加快病毒文件的判断，减少客户端和服务器之间的通信，提高工作效率，降低对移动终端的性能影响。

表1 危险行为示例

系统文件	程序文件	用户文件
修改	调用	读取
删除	修改	修改
重命名	删除	删除
调用功能模块	重命名	重命名

危险行为的引入，不仅减少了客户端对服务器的访问，降低了系统通信代价，而且加快了病毒文件的判断，提高了服务器的负载能力。

4.4 规则制定

规则的定义和完善的规则是云监控模型能否防御病毒入侵的关键因素。

规则定义很简单，由文件标识 f、与文件相对应的规则 n 组成，可表示为二元组 (f,n)。

下面按照文件 $f\in F'$、$f\notin F'$但 $f\in F$ 和 $f\notin F$ 来分别讨论相应访问规则的完善。

对于 $f\in F'$，直接从规则列表中按照文件 f 相对应的访问规则 (f,n) 进行监控即可。

对于 $f\notin F'$但 $f\in F$，由客户端向服务器上的规则库查询与文件 f 相对应的访问规则 (f,n)，下载到规则列表中，然后据此进行监控。

对于 $f\notin F$，当客户端第一次向规则库查询时，规则库返回默认访问规则 (f,n_1)，其中 $n_1=$｛进行文件性质判断｝，且此规则只生效一次，当下一次客户端进行查询时，如果系统已经分析出文件 f 的性质并制定了相对应的访问规则 (f,n')，那么返回新的规则 (f,n')，否则依然返回默认规则 (f,n_1)，且只生效一次。

对于文件为已知的正常文件或者病毒文件，由服务器智能设定或者由后台专业人士进行具体设定。智能设定的规则为：对于正常文件，不进行监控，减少对移动终端的资源消耗；对于病毒文件，阻止运行，确保不威胁移动终端安全。后台工作人员则按照文件的具体特点进行针对性设定。

对于文件为移动终端上可以与外界发生信息交互的蓝牙、网络模块等，都设定为需要进行监控，以配合系统监测的被动监控模式。

规则的制定可以由移动终端用户自行设定，但只存储于规则列表，不会被服务器的规则库所收录，这样在为用户提供自主研究空间的同时保证了规则中访问规则的安全性。

5 云监控安全性分析

云监控的功能在于按照文件访问规则监控文件的运行，它的安全性保证在于：所有文件都有访问规则；所有需要监控的文件都被监控。下面分别进行形式化证明。

5.1 所有文件都有访问规则

所有文件都有访问规则，可以形式化表示为：对 $\forall f_i$，都 $\exists n_j\in N$，使得

$(f_i,n_j)=Q(f_i)$

证明：假设 $\exists f_i$，对 $\forall n_l\in N$，使得

$(f_i,n_l)\neq Q(f_i)$

当 $f_i\in F'$时，

$Q(f_i)=(f_i,n_k),n_k\in N$

当 $f_i\notin F'$，而 $f_i\in F$时，

$Q(f_i)=(f_i,n_m),n_m\in N$，且 $m\neq 1$

当 $f_i\notin F$时，f_i 为新的文件，当客户端第一次向规则库查询时：

$Q(f_i)=(f_i,n_1)$

当客户端第二次向规则库查询时：

$Q(f_i)=(f_i,n_p)=h((f_i,k(f_i)))$

其中 n_p 为规则制定模块根据 f_i 的文件性质制定的新规则，此时 $N=N\cup\{n_p\}$即 $n_p\in N$。

上述讨论包含了所有的情况，即总是 $\exists n\in N$，使得

$(f_i,n)=Q(f_i)$

与假设 $\exists f_i$，对 $\forall n_l\in N$，使得 $(f_i,n_l)\neq Q(f_i)$矛盾。即，对 $\forall f_i$，都 $\exists n_j$，使得 $(f_i,n_j)=Q(f_i)$的结论成立，云监控模型能够保证所有文件都有访问规则。

5.2 所有需要监控的文件都被监控

由4.2节系统监测可知，在规则列表中所有需要进行监控的文件都将带动系统监测模块的运行，也就是说规则列表中规定需要监控的文件都将被监控。那么只要证明规则列表中包含了所有移动终端系统上需要监控的文件。

假设移动终端上存在文件 $f_i\notin F'$，那么当 $f_i\in F$时，规则列表可以从规则库获取文件 f_i 的访问规则；当 $f_i\notin F$时，规则列表也可以从规则库获取到文件 f_i 的默认访问规则 (f_i,n_1)。经过一轮的更新操作，将使得 $f_i\in F'$，因此规则列表中能够包含所有移动终端系统上需要监控的文件，即所有需要监控的文件都被监控。

通过5.1节和5.2节的证明，可知云监控模型中“所有文件都有访问规则”、“所有需要监控的文件都被监控”的结论是成立的，也就是说云监控模型能够为移动终端提供安全的监控服务。

结束语 以云安全技术对HIPS进行改进，形成本文的云监控模型。客户端不需要存储大量的访问规则，不需要进行定期的规则更新，降低了对移动终端的存储空间占用；运用单步危险行为分析方法来判断文件性质，加快了病毒文件的判断，降低了客户端与服务器之间的通信代价；放弃传统驻留监控的思路，由主动监控转变为被动监控，降低了对移动终端的性能影响，提高了系统监测的工作效率，能够满足移动终端对轻量级、高效率客户端的要求。云监控模型仍然存在一定的问题，如基于MD5值的黑白名单技术过于严格，使得黑白名单的更新较为频繁，需要在后续的工作中进行改进。病毒防治只是移动终端安全问题的一个方面，对本文提出的云监控模型进行仿真实验，检验云监控的具体工作性能，并在现有基础上进行功能深化和扩展，是今后的研究方向和工作重点。

（下转第74页）

其中，查询语句给出攻击者与实体 A 同时获得共享密钥 Kab 的不安全终止状态。同样可以查询攻击者与实体 A 同时获得其他共享密钥的不安全终止状态。

查询结果如下：

下面以功能单元 1 为例分析不安全状态及其攻击路径。其中对通过 on-the-fly 生成的攻击路径及采用 TMN 协议的执行过程进行了描述，S_{ij} ($i=1,2;j=1,2,3,4$)代表第 i 次会话的第 j 步。

·初始者 A 共享密钥：Kab；响应者 B 共享密钥：Kab；攻击者获得共享密钥：Kab。

$S_{11}.\ A\rightarrow J:(B,ENC_{K_{jp}}(K_{aj})),A$

$S_{12}.\ J\rightarrow B:A$

$S_{13}.\ B\rightarrow J:(A,ENC_{K_{jp}}(K_{ab})),B$

$S_{14}.\ J\rightarrow A:B,ENC_{K_{aj}}(K_{ab})$

$S_{21}.\ In\rightarrow J:(B,ENC_{K_{jp}}(K_i)),A$

$S_{22}.\ J\rightarrow In:A$

$S_{23}.\ In\rightarrow J:(A,ENC_{K_{jp}}(K_{ab})),B$

$S_{24}.\ J\rightarrow In:B,ENC_{K_i}(K_{ab})$

在功能单元 5 中获得了在文献[4,7]没有提到的新的攻击模式对应的攻击路径如下：

$S_{11}.\ A\rightarrow J:(B,ENC_{K_{jp}}(K_{aj})),A$

$S_{21}.\ In\rightarrow J:(B,ENC_{K_{jp}}(K_i)),A$

$S_{22}.\ J\rightarrow In:A$

$S_{12}.\ J\rightarrow B:A$

$S_{13}.\ B\rightarrow J:(A,ENC_{K_{jp}}(K_{ab})),B$

$S_{23}.\ In\rightarrow J:(A,ENC_{K_{jp}}(K_{ab})),B$

$S_{24}.\ J\rightarrow In:B,ENC_{K_i}(K_{ab})$

$S_{14}.\ J\rightarrow A:B,ENC_{K_{aj}}(K_{ab})$

结束语 随着密码协议的广泛应用，它的正确性和安全性越来越受到关注。形式化建模分析是验证密码协议的一种有效方法。以往的工作大多集中在对密码协议的单会话过程进行分析，本文使用有色 Petri 网对密码协议多次并发会话进行形式化建模和验证，并以 TMN 协议为例进行形式化分析，发现该协议具有多个攻击路径以及新的攻击模式。由于不断增加的密码协议的复杂性以及安全属性的多样性，今后课题组主要针对分析和验证新的安全性质，如非否认性、匿名性、公平性等开展研究工作。

参 考 文 献

[1] Jensen K, Kristensen L, Wells L. Coloured Petri Nets and CPN Tools for modelling and validation of concurrent systems[J]. International Journal on Software Tools for Technology Transfer (STTT), 2007, 9(3): 213-254

[2] Dolev D, Yao A. On the security of public key protocols. Information Theory[J]. IEEE Transactions on Information Theory, 1983, 29(2): 98-208

[3] Tatebayashi M, Matsuzaki N, Jr D B. Key distribution protocol for digital mobile communication systems[M]. Springer-Verlag, 1989: 324-333

[4] Yu-Qing Z, Xiu-Ying L. An Approach to the Formal Analysis of TMN Protocol[M]. Progress on Cryptography, 2004: 235-243

[5] Lowe G, Roscoe B. Using CSP to detect errors in the TMN protocol[J]. Software Engineering, IEEE Transactions on Software Engineering, 1997, 23(10): 659-669

[6] 薛锐，冯登国. 安全协议的形式化分析技术与方法[J]. 计算机学报，2006，29(1)：1-20

[7] Permpoontanalarp Y. On-the-Fly Trace Generation and Textual Trace Analysis and Their Applications to the Analysis of Cryptographic Protocols [M]. Formal Techniques for Distributed Systems, 2010: 201-215

[8] Al-Azzoni I, Down D, Khedri R. Modeling and verification of cryptographic protocols using coloured petri nets and design/CPN[J]. Nordic Journal of Computing, 2005, 12(3): 201-228

[9] 黎波涛，罗军舟. 不可否认协议的 Petri 网建模与分析[J]. 计算机研究与发展，2005，42(9)：1571-1577

[10] Tritilanunt S, Boyd C, Foo E. Using Coloured Petri Nets to Simulate Dos-resistant Protocols[C]// Proceeding of 7th Workshop and Tutorial on Practical Use of Coloured Petri Nets and the CPN Tools. 2006: 261-280

[11] Lee G, Lee J. Petri Net Based Models for Specification and Analysis of Cryptographic Protocols[J]. The Journal of System and Software, 1997, 37: 141-159

[12] Dresp W. Security Analysis of the Secure Authentication Protocol by Means of Coloured Petri Nets[C]// Proceeding of 9th IFIP Communication and Multimedia Security. 2005

[13] Permpoontanalarp Y, Changkhanak A. Security analysis of the TMN protocol by using Coloured Petri Nets: On-the-fly trace generation method and homomorphic property[C]// Computer Science and Software Engineering (JCSSE), 2011 Eighth International Joint Conference. 2011: 63-68

(上接第 58 页)

参 考 文 献

[1] 陈建民. 3G 时代手机病毒的威胁与移动安全[J]. 信息网络安全，2009(09)：19-20

[2] 朱圣军，刘功申，罗俊，等. 智能手机病毒与信息安全[J]. 信息安全与通信保密，2011(05)：96-100

[3] Grant. CNCERT 安全报告：软件漏洞成重大隐患[J]. 网络与信息，2011(04)：57

[4] 马云雷，刘功申，葛克为，等. 基于 Mobile 的手机杀毒软件设计与实现[J]. 信息技术，2011(01)：7-80

[5] 王磊，张玉清. WM 平台下反病毒软件的设计与实现[J]. 计算机工程，2009(21)：144-150

[6] 吴俊军，方明伟，张新访. 基于启发式行为监测的手机病毒防治研究[J]. 计算机工程与科学，2010(01)：35-38

[7] 杨建强，吴钊，李学锋. 增强智能手机安全的动态恶意软件分析系统[J]. 计算机工程与设计，2010(13)：2969-2971

[8] 刘鹏. 云计算[M]. 北京：电子工业出版社，2010：1-3

[9] 杨文志. 云计算技术指南：应用、平台与架构[M]. 北京：化学工业出版社，2010：7-12

[10] 瑞星云安全计划白皮书[EB/OL]. http://sec.chinabyte.com/113/8544113.shtml, 2011-4-20

[11] 趋势科技云安全白皮书[EB/OL]. http://sec.chinabyte.com/458/8546458.shtml, 2011-4-20

[12] 杨磊. 主机入侵防御系统的应用[J]. 计算机安全，2005(4)：20-22